78-FM-30
Vol. X, Rev. 1

JSC-14266

OPS MCC Ground Navigation Program

Level C

Orbit Determination Processing

Formulation Requirements

State Transition Matrix Module (STMM)

Mission Planning and Analysis Division

September 1979

NASA
National Aeronautics and Space Administration

Lyndon B. Johnson Space Center
Houston, Texas

78FM30:X

78-FM-30
Vol. X
Rev. 1

JSC-14266

SHUTTLE PROGRAM

OPS MCC GROUND NAVIGATION PROGRAM
LEVEL C ORBIT DETERMINATION PROCESSING

FORMULATION REQUIREMENTS
STATE TRANSITION MATRIX MODULE (STMM)

By W. P. Girod
TRW Defense and Space Systems Group

JSC Task Monitor: J. B. Williamson, Mathematical Physics Branch

Approved: Wilbur R. Wollenhaupt
Wilbur R. Wollenhaupt, Head
Ground Navigation Section

Approved: E. R. Schiesser
Emil R. Schiesser, Chief
Mathematical Physics Branch

Approved: [illegible]
Ronald L. Berry, Chief
Mission Planning and Analysis Division

Mission Planning and Analysis Division

National Aeronautics and Space Administration

Lyndon B. Johnson Space Center

Houston, Texas

September 1979

PREFACE

The Mathematical Physics Branch/Mission Planning and Analysis Division has the responsibility to provide the functional ground navigation software formulation requirements for the Mission Control Center (MCC) low-speed-processing phases during Operations Project Shuttle (OPS).

The ground navigation software formulation requirements are logically organized into volumes. This organization is presented in the accompanying table. The material in each volume presents the level C formulation requirements of the processors and modules required to process low-speed-tracking data and perform orbit determination and other related navigation computations. Each volume describes the formulation requirements of the identified processor or module specified in the OPS MCC Ground Navigation Program Level B Software document (ref. 1). The inputs and outputs required to accomplish the functions described are specified. Flow charts defining the sequence of mathematical operations and display and control processing required to satisfy the described functions are included in the document where appropriate.

OPS MCC GROUND NAVIGATION PROGRAM LEVEL C SOFTWARE REQUIREMENTS

ORBIT DETERMINATION PROCESSING FORMULATION DOCUMENT

VOLUME X

STATE TRANSITION MATRIX MODULE (STMM)

CONTENTS

ACRONYMS AND ABBREVIATIONS

BB	batch-to-batch (DC processing mode)
CMP	covariance matrix processor
DC	differential correction
DCM	differential correction module
I/F	interface
M50	mean of 1950 coordinate system
SB	superbatch (DC processing mode)
STMM	state transition matrix module

Physical Units

ft	international feet
E.r.	Earth radii
sec	SI seconds (second of atomic time in the international system)
hr	hours (3600 sec)
lb	pounds (used to denote both pound-force and pound-mass)
slg	slugs (mass unit in ft-lb-sec dynamic system of units)
rad	radians

1.0 CORRELATION TO LEVEL B

This document contains the level C software requirements for the state transition matrix module (STMM). Functionally, the output of the STMM is that called for in the level B requirements (ref. 1). The internal structure and methods of computation, however, have undergone major revision.

2.0 GENERAL DESCRIPTION OF STMM

The STMM computes partial derivatives of a specified vehicle state (6 dimension M50 Cartesian state) with respect to a specified anchor state (6 + M dimension M50 Cartesian vector plus M dynamic parameters). Dynamic parameters are constants that appear in certain perturbing force models and therefore affect the trajectory. The solution vector in orbit determination is the ordered set of parameters that are estimated in the differential correction (DC) process. In the Space Shuttle Ground Navigation Program, this set consists of the M50 Cartesian state (vehicle position and velocity) at anchor time t_o, the dynamic parameters, and radar observation biases. Since observation biases do not affect the vehicle trajectory, partial derivatives of the vehicle state with respect to them are trivial and are not included in the STMM computations.

The solution vector order in this application is as follows.

a. X_o = M50 Cartesian state (position and velocity) at epoch t_o

b. α = ordered set of dynamic parameters. These parameters are limited to the components (in vehicle body coordinates) of up to three sets of vent forces and the drag constant multiplier. The defined order of these parameters are:

$\alpha_{V1} = \{\alpha_{V1}(X_b), \alpha_{V1}(Y_b), \alpha_{V1}(Z_b)\}$, components of vent-1 force in vehicle body coordinates

$\alpha_{V2} = \{\alpha_{V2}(X_b), \alpha_{V2}(Y_b), \alpha_{V2}(Z_b)\}$

$\alpha_{V3} = \{\alpha_{V3}(X_b), \alpha_{V3}(Y_b), \alpha_{V3}(Z_b)\}$

α_D = drag multiplier

c. b = ordered set of observation biases. (Precise definition and order of this not relevant in the STMM.)

The total number of solution vector parameters is limited to 15. The first six elements, X_o, are always present in the solution vector. The remaining elements (up to nine) are specified from the dynamic parameter and observation bias subsets; however, the relative order of the solution vector elements remains as defined above.

The partial derivative matrices computed by the STMM are $T(t,t_o) = \partial X/\partial X_o$ and $P(t,t_o) = \partial X/\partial \alpha$ where X is the vehicle state at time t. The STMM output is the composite 6X(6+M) matrix $\phi(t,t_o) = \left(T(t,t_o) | P(t,t_o)\right)$.

The partial derivative $T = \partial X/\partial X_o$ may be computed (to a good approximation) in one step via a mean conic reference orbit defined by the endpoint states (X_o, t_o) and (X, t). The partial derivative $P = \partial X/\partial\alpha$ is more difficult to obtain, in that some type of numerical integration must be performed.

In order to compute these partial derivatives in the most efficient manner, the STMM uses two modes of operation.

a. Mode 1 is used for 6x6 covariance propagation for display, and for construction of a priori covariance matrices for DC solutions.

b. Mode 2 is used for propagation of the superbatch (SB) solution covariance to the SB end-time, and for the computation of measurement partial derivatives in DC cases.

Mode 2 operates in response to calls from the Differential Correction Module (DCM) and the Convergence Processor (CP). Mode 1 operates in response to calls from the Covariance Matrix Processor (CMP).

The STMM is composed of two major sections.

a. Mean conic states partial derivative (mode 1).- This function is called with two 6-element Cartesian state vectors and their epochs X_A, t_A and X_B, t_B. It computes the Cartesian state transition matrix $T(B,A) = X/_B/X_A$ in one step using a mean conic reference orbit defined by the the two input states. This function is exercised by the state transition integrator, described below, when the STMM operates in mode 2.

b. State transition integrator (mode 2).- In mode 2 the STMM output may contain current state partial derivatives with respect to each of the dynamic parameters (vent forces and drag multiplier) in the solution vector. These derivatives require numerical integration of some form of the state variational equations. The state transition integrator provides a simple numerical integration technique for evaluating these derivatives.

   In this mode the STMM inputs are the parameters required to initialize the integration. The initial and final states for the integration step consists of a 6 x 6 + M matrix of partial derivatives $\left(T(t,t_o)\middle|P(t,t_o)\right)$ where $T = \partial X/\partial X_o$ and $P = \partial X/\partial\alpha$, with $\alpha = (\alpha_1 \cdots \alpha_M)$ representing the ordered set of M dynamic parameters in the solution vector.

   This output is generated by the state transition integrator, which computes $T(t,t_o)$ in a stepwise fashion using the multiplicative property of the Cartesian state transition matrix.

3.0 DETAILED FUNCTIONAL REQUIREMENTS FOR THE STMM

The STMM utilizes two modes of operation to satisfy various user requirements for state transition matrices. The requirements for these modes are given in sections 3.1 and 3.2.

3.1 MODE 1: MEAN CONIC STATE PARTIAL DERIVATIVES

The inputs to the STMM in this mode are two M50 Cartesian states and their epochs, X,t and X_o,t_o. The output is the 6x6 state transition matrix $T(t,t_o) = \partial X/\partial X_o$, which is computed in one step via a mean conic reference orbit defined from the two endpoint states. This mode is utilized by the CMP for several of its applications. It also serves as a subfunction for the DC mode (sec. 3.2) of the STMM.

The computational requirements (refs. 4 and 5) for obtaining $T(t,t_o)$ are identical to those in the OFT Ground Navigation Program, with the computation of $\partial X/\partial\mu$ removed (μ = Earth gravitational parameter). Computational requirements are given here for completeness, but this does not imply that the verified coding in the OFT program should be altered (except for the deletion of $\partial X/\partial\mu$).

In the following computational requirements, the input states are used in the form $X = \{R,\dot{R}\}$, $X_o = \{R_o,\dot{R}_o\}$.

Inputs:

$R,\dot{R},t$	M50 cartesian state at time t
$R_o,\dot{R}_o,t_o$	M50 cartesian state at time t_o

Constants: μ = Earth gravitational parameter

$$r = (R \cdot R)^{1/2} \quad ; \quad r_o = (R_o \cdot R_o)^{1/2}$$

$$V^2 = \dot{R} \cdot \dot{R} \quad ; \quad V_o^2 = \dot{R}_o \cdot \dot{R}_o$$

$$D = R \cdot \dot{R} \quad ; \quad D_o = R_o \cdot \dot{R}_o$$ (Note: Dimension R, $\dot{R}$, R_o and $\dot{R}_o$ are 3x1.)

$$\alpha = 1/2\left[\frac{2\mu}{r} - V^2 + \frac{2\mu}{r_o} - V_o^2\right] \tag{1}$$

$$\Psi = \frac{\alpha}{\mu}(t-t_o) + \frac{1}{\mu}(D-D_o)$$

$$\chi^2 = \alpha\Psi^2$$ (Note: χ^2 is just the name of a parameter; there will be no χ. Compute as $\chi^2 = (\alpha\Psi)\Psi$.)

If $|\chi^2| > 1$, $\chi^2_1 = 1/4\ \chi^2$

If $|\chi^2_1| > 1$, $\chi^2_2 = 1/4\ \chi^2_1$

.

.

.

Continue until $|\chi^2_m| \leq 1$ (2)

$$C_5(m) = 1/5! \left[1-\left(1-\left(1-\left(1-\left(1-\left(1-\left(1-\left(1-\frac{\chi_m^2}{21*20}\right)\frac{\chi_m^2}{19*18}\right)\frac{\chi_m^2}{17*16}\right)\frac{\chi_m^2}{15*14}\right)\frac{\chi_m^2}{13*12}\right)\frac{\chi_m^2}{11*10}\right)\frac{\chi_m^2}{9*8}\right)\frac{\chi_m^2}{7*6}\right]$$

$$C_4(m) = 1/4! \left[1-\left(1-\left(1-\left(1-\left(1-\left(1-\left(1-\left(1-\frac{\chi_m^2}{20*19}\right)\frac{\chi_m^2}{18*17}\right)\frac{\chi_m^2}{16*15}\right)\frac{\chi_m^2}{14*13}\right)\frac{\chi_m^2}{12*11}\right)\frac{\chi_m^2}{10*9}\right)\frac{\chi_m^2}{8*7}\right)\frac{\chi_m^2}{6*5}\right]$$

$$C_3(m) = 1/3! - \chi_m^2\, C_5(m)$$

$$C_2(m) = 1/2! - \chi_m^2\, C_4(m)$$ (3)

$$C_1(m) = 1 - \chi_m^2\, C_3(m)$$

$$C_0(m) = 1 - \chi_m^2\, C_2(m)$$

If $m \neq 0$, compute

$C_0(m-1) = 2(C_0(m))^2-1$

$C_1(m-1) = C_0(m)\ C_1(m)$

.

. Perform this process a total of m times (decreasing m by one on each step) (4)

.

$C_o = 2(C_0(1))^2-1$

$C_1 = C_0(1)\ C_1(1)$

$C_2 = (1-C_o)/\chi^2$

$C_3 = (1-C_1)/\chi^2$

$C_4 = (1/2!-C_2)/\chi^2$ (5)

$C_5 = (1/3! - C_3)/\chi^2$

If $m = 0$, $C_i = C_i(m)$, $i = 0 , \ldots , 5$ (from eq. (3))

$S_1 = \Psi C_1$

$S_2 = \Psi(\Psi C_2)$ (6)

$S_3 = \Psi(\Psi(\Psi C_3))$

$U = S_2(t-t_o) + \mu\Psi^5 (C_4 - 3C_5)$

$(f-1) = -\frac{\mu}{r_o} S_2 \quad ; \quad f = 1 + (f-1)$

$$g = t - t_o - \mu S_3 \qquad \text{(Note: } g = r_o S_1 + D_o S_2 \text{ may also be used.)} \tag{7}$$

$$\dot{f} = -\frac{\mu}{r r_o} S_1$$

$$(\dot{g}-1) = -\frac{\mu}{r} S_2 \quad ; \quad \dot{g} = 1 + (\dot{g}-1)$$

$$\ddot{R}_o = -\frac{\mu}{r_o^3} R_o \quad ; \quad \ddot{R} = -\frac{\mu}{r^3} R \quad ; \quad I_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

$$\frac{\partial R}{\partial R_o} = -\left[\frac{\dot{f}\, S_1}{r_o} + \frac{(f-1)}{r_o^2}\right] R\, R_o^T - \dot{f} S_2\, R\dot{R}_o^T$$

$$+ \frac{(f-1)S_1}{r_o} \dot{R}R_o^T + (f-1)\, S_2\, \dot{R}\dot{R}_o^T$$

$$+ U\, \dot{R}\ddot{R}_o^T + f\, I_3$$

$$\frac{\partial R}{\partial \dot{R}_o} = -\dot{f} S_2\, RR_o^T - (\dot{g}-1)\, S_2\, R\dot{R}_o^T + (f-1)\, S_2\, \dot{R}R_o^T$$

$$+ (gS_2 - U)\, \dot{R}\dot{R}_o^T + g\, I_3 \tag{9}$$

$$\frac{\partial \dot{R}}{\partial R_o} = -\dot{f}\left[\frac{C_o}{r r_o} + \frac{1}{r^2} + \frac{1}{r_o^2}\right] RR_o^T$$

$$-\left[\frac{\dot{f}S_1}{r} + \frac{(\dot{g}-1)}{r^2}\right] R\dot{R}_o^T$$

$$+ \left[\frac{\dot{f} S_1}{r_o} + \frac{(f-1)}{r^2} \right] \dot{R} R_o^T$$

$$+ \dot{f} S_2 \dot{R} \dot{R}_o^T + U \ddot{R} \ddot{R}_o^T + \dot{f} I_3$$

$$\frac{\partial \dot{R}}{\partial \dot{R}_o} = - \left[\frac{\dot{f} S_1}{r} + \frac{(\dot{g}-1)}{r^2} \right] R R_o^T$$

$$- \frac{(\dot{g}-1) S_1}{r} R \dot{R}_o^T + \dot{f} S_2 \dot{R} R_o^T$$

$$+ (\dot{g}-1) S_2 \dot{R} \dot{R}_o^T - U \ddot{R} \dot{R}_o^T + \dot{g} I_3$$

Assemble the 6x6 state transition Matrix for return to user.

$$T(t,t_o) = \left[\begin{array}{c|c} \partial R/\partial R_o & \partial R/\partial \dot{R}_o \\ \hline \partial \dot{R}/\partial R_o & \partial \dot{R}/\partial \dot{R}_o \end{array} \right] \tag{10}$$

Users of this function: CMP (also used by STMM as a subfunction)

Inputs	Outputs
X, t, X_o, t_o	$T(t, t_o)$

Interfaces:

I/F function	Inputs to I/F	Outputs from I/F
System parameters	--	μ

3.2 MODE 2: STATE TRANSITION INTEGRATOR

This mode of the STMM is exercised by the DCM for cases in which state partial derivatives with respect to dynamic parameters may be required. The method of computation given here (stepwise computation of $\partial X/\partial X_o$ coupled with a simple numerical integration for $\partial X/\partial\alpha$) is described in references 2 and 3.

3.2.1 Computation of State Transition Matrix

The DCM will request a state transition matrix $\phi(t,t_o) = \left[T(t,t_o)|P(t,t_o)\right]$ at a specified output time t. In order to effect this computation, the DCM will provide sufficient information to initialize the STMM at an initial time t_L. The STMM then updates the initial matrix $\phi(t_L,t_o)$ (in one computational step) to time t, and returns the output matrix $\phi(t,t_o)$ to the DCM. No vehicle ephemeris is required by the STMM for this computation. The computation procedure is as follows.

Inputs

a. X,t = M50 state and epoch at output time

b. X_L,t_L = Initial M50 state and epoch

c. $\phi(t_L,t_o) = \left[T(t_L,t_o)|P(t_L,t_o)\right]$, state transition matrix at initial time t_L

d. Solution vector content = Identifies which dynamic parameters are present in the DC solution (let M = total number of dynamic parameters)

e. VNT_J = Flag(s) that specify whether J-th vent is ON or OFF (if vents are included in solution)

   VNT_J = ON means J-th vent is ON (or t_L is the ON time)

   VNT_J = OFF means J-th vent is OFF (or t_L is the OFF time)

f. Link ID

g. Integrator force options (for determination of drag model)

The computational steps are as follows.

a. Compute $T(t,t_L)$ with mean conic state partial derivative function of section 3.1

   $T(t,t_L)$ Compute per section 3.1

b. Compute 6x6 cartesian state transition matrix.

$$T(t,t_o) = T(t,t_L)\ T(t_L,t_o)$$

c. Compute the 3xM dynamic parameter partial derivative matrices via the procedure of section 3.2.2.

$$\frac{\partial A_L}{\partial \alpha_L}, \frac{\partial A_t}{\partial \alpha_t} \text{ Compute per section 3.2.2.}$$

d. Compute the 6xM matrix $P(t,t_L)$ using the trapezoid rule.

$$P(t,t_L) = \frac{t-t_L}{2}\left[T(t,t_L)\begin{bmatrix} 0 \\ 3xM \\ \hline \partial A_L/\partial \alpha_L \end{bmatrix} + \begin{bmatrix} 0 \\ 3xM \\ \hline \partial A_t/\partial \alpha_t \end{bmatrix}\right]$$

e. Compute $P(t,t_o)$ from

$$P(t,t_o) = T(t,t_L)\ P(t_L,t_o) + P(t,t_L)$$

f. The output of the 6 x (6 + M) state transition matrix is

$$\phi(t,t_o) = \left(T(t,t_o) \,|\, P(t,t_o)\right)$$

(Note: Appendix A gives a brief outline of the rationale for the computations of this section.)

3.2.2 Computation of Acceleration Partial Derivatives

The computation of the state transition matrix (sec. 3.2.1) requires acceleration partial derivative matrices $\partial A/\partial \alpha$ at the two endpoints of the current computation step. The procedure for obtaining these matrices is as follows.

a. Vent partial derivatives.- The procedure for computing the 3x3 partial derivative matrices $\partial A/\partial \alpha_{VJ}$ for each vent is as follows. For J-th vent

If VNT_J = OFF: $\partial A/\partial \alpha_{VJ} = (0)_{3x3}$

If VNT_J = ON: Compute $\partial A/\partial \alpha_{VJ}$ as follows:

From vehicle weight table, obtain

W(t) Vehicle mass at time t

From vehicle attitude table, obtain

B(t) = Transformation (3x3) from body to M50 at t

From system parameters obtain

$$(GVENT) = \text{Conversion factor } \frac{\text{lb E.r./hr}^2}{\text{slg ft/sec}^2}$$

(Typical value is 19.9264496203518)

$$\frac{\partial A}{\partial \alpha_{VJ}} = \frac{(GVENT)}{W(t)} B(t)$$

This computation (for each vent) is performed for the two times t_L and t required in section 3.2.1.

b. Drag partial derivatives.- The procedure for computing the 3x1 partial derivative matrix $\partial A/\partial \alpha_D$ for the drag multiplier is as follows.

Compute the value of the ballistic coefficient

$$\beta(t) = \frac{K_D \ C_D \ S_A \ g_o}{2 \ W(t)}$$

K_D = Link dependent value of drag multiplier (link dependent parameter)

C_D = Nominal value of drag constant (system parameter)

S_A = Reference area of vehicle $(ft)^2$

g_o = Standard gravity (lb/slg) (system parameter: typical value is 32.174048556)

W(t) = Vehicle mass (lb)

Assemble inputs required to exercise density module

$\vec{r} = (x,y,z)^T$, vehicle position (M50) at desired time t.

$\vec{r}(SUN)$ = Solar position vector (M50) at time t obtained from solar ephemeris

From density module, obtain atmospheric density of vehicle at time t

$\rho(t)$ = Atmospheric density (slg/ft^3)

Compute Earth spin vector in M50 coordinates

$$\vec{\omega}_E = (RNP)^T \begin{bmatrix} 0 \\ 0 \\ \omega_E \end{bmatrix}$$

(RNP) = Transformation (3x3) from M50 to TEI (see volume XIV of these requirements for definition)

ω_E = Earth rate (rad/hr) (system parameter)

From vehicle position and velocity ($\vec{r}$, $\vec{v}$, in M50 at time t), compute $\vec{V}_A$, the velocity relative to the atmosphere

$$\vec{V}_A = \vec{V} - \vec{\omega}_E \times \vec{r}$$

$$V_A = |\vec{V}_A|$$

Compute drag acceleration

$$\vec{A}_D = - \beta(t)\ \rho(t)\ V_A\ \vec{V}_A$$

Note that $\vec{A}_D$ is in mixed units, but that the partial derivative computed in the next step will result in the proper internal units.

Compute drag partial derivative matrix

$$\frac{\partial A}{\partial \alpha_D} = A_D*(FEET)$$

(FEET) = System parameter in units of ft/E.R.
(typical value is 20.92573819×10^6)

c. Construct the 3xM partial derivative matrix $\left(\partial A/\partial \alpha\right)$ for time t. Append the vent matrices $\frac{\partial A}{\partial \alpha_{VJ}}$ and the drag matrix $\frac{\partial A}{\partial \alpha_D}$ in solution vector order.

Example - If the solution vector content identifies vent 1, vent 2, and drag as the dynamic parameters in the solution

$$\frac{\partial A}{\partial \alpha} = \left[\frac{\partial A}{\partial \alpha_{V1}} \;\vdots\; \frac{\partial A}{\partial \alpha_{V2}} \;\vdots\; \frac{\partial A}{\partial \alpha_D} \right]$$

INTERFACES

I/F function	Input to function	Output from function
Vehicle weight table	t	W(t)
Vehicle attitude table	t	B(t)
Vehicle profiles Coefficient of drag Variable drag constants	t, drag option	$C_D(t)$, S_A , K_D
Solar ephemeris	t	$\vec{R}$(SUN)
Density module	$\vec{r}, \dot{\vec{r}}$,(SUN),t	$\rho(t)$
Systems parameters		μ,(GVENT),g_o,(RNP), ω_E,(FEET)

4.0 INPUTS

Mode 1: (user is CMP).

X, t = M50 state and epoch at desired output time

X_o, t_o = M50 state and epoch at initial time

Mode 2: (user is DCM or CP)

X, t = M50 state and epoch at output time

X_L, t_L = Initial M50 state and epoch

$\phi(t_L, t_o) = [T(t_L, t_o) | P(t_L, t_o)]$, state transition matrix at initial time t_L

Solution vector content = Identifies which dynamic parameters are present in the DC solution (let M = total number dynamic parameters)

VNT_J = Flag(s) that specify if J-th vent is ON or OFF (if vents are included in solution)

VNT_J = ON means J-th vent is ON (or t_L is the ON time)

VNT_J = OFF means J-th vent is OFF (or t_L is the OFF time)

Link ID

Integrator force options (for determination of drag model)

5.0 OUTPUTS

Mode 1: (user is CMP).

$T(t, t_o)$ = State transition matrix (6x6) from t_o to t.

Mode 2: (user is DCM or CP)

$\phi(t, t_o) = [T(t, t_o) | P(t, t_o)]$, state transition matrix, 6 x (6 + M), from t_o to t.

6.0 CONSTRAINTS

All numerical computations in the STMM are in double precision.

7.0 REFERENCES

1. Level B Software: Preliminary Orbit Determination Processing Formulation Requirements. JSC IN 77-FM-57, Oct. 1977.

2. Goodyear, W. H.: A Quadrature Formula for Parameter Partials. AIAA Astrodynamics Conference, Palo Alto, California, August 1978.

3. Goodyear, W. H.: An Extension of the Present Analytic Approximation for the State Transition Matrix to an Approximation for the Parameter Partial Matrix. Informal paper distributed to Mission Planning and Analysis Division, JSC.

4. Goodyear, W. H.: A General Method for the Computation of Cartesian Coordinates and Partial Derivatives of the Two-Body Problem. NASA CR-522, Sept. 1966.

5. Goodyear, W. H.: Completely General Closed-Form Solution for Coordinates and Partial Derivatives of the Two-Body Problem. Astronomical Journal, V. 70, No. 3, April 1965.

6. Space Shuttle Astrodynamical Constants. JSC IN 78-FM-32, June 1978.

7. Coordinate System Standards in the Space Shuttle Program. NASA TM X-58153, Oct. 1974.

8. JSC-11173, OFT MCC Level B & C Requirements for the Free Flight Predictor", MPAD (76-FM-15), May 13, 1977.

APPENDIX A

STATE TRANSITION QUADRATURE APPROXIMATION

APPENDIX A

STATE TRANSITION QUADRATURE APPROXIMATION

This appendix gives a brief discussion of the methods presented in references 3 and 4 that are used in the state transition integrator (sec. 3.2).

Let X represent the six-dimensional state of cartesian position and velocity elements of a material body that is subject to known noncentral-body forces. The equations of motion of this body have the general form

$$\frac{dX}{dt} = \dot{X}(X(X_o, t_o, \alpha, t), \alpha, t)$$

X_o = Initial conditions at time t_o

α = Multiple of dynamic parameters that determine the magnitude of noncentral-body forces acting on the body

t = Time (independent variable)

The differential equation for the partial derivatives $\partial X/\partial X_o$ has the form

$$\frac{d}{dt}\left(\frac{\partial X}{\partial X_o}\right) = \frac{\partial \dot{X}}{\partial X} \frac{\partial X}{\partial X_o} \tag{A1}$$

The differential equation for the partial derivative of X with respect to the dynamic parameter α has the following form (use chain rule for partial derivatives).

$$\frac{d}{dt}\left(\frac{\partial X}{\partial \alpha}\bigg|_{t_o}\right) = \frac{\partial \dot{X}}{\partial X} \frac{\partial X}{\partial \alpha}\bigg|_{t_o} + \frac{\partial \dot{X}}{\partial \alpha}\bigg|_{t} \tag{A2}$$

$\frac{\partial}{\partial \alpha}\Big|_{t_o} \rightarrow \alpha$ and X_o considered independent (X_o fixed)

$\frac{\partial}{\partial \alpha}\Big|_{t} \rightarrow \alpha$ and X considered independent (X fixed)

In the physical problems of interest, the solutions to equation (A1) and equation (A2) have the transitive properties

$$\frac{\partial X}{\partial X_o} = \frac{\partial X}{\partial X_1}\frac{\partial X_1}{\partial X_o} \tag{A3}$$

$$\left.\frac{\partial X}{\partial \alpha}\right|_{t_o} = \frac{\partial X}{\partial X_1}\left.\frac{\partial X_1}{\partial \alpha}\right|_{t_o} + \left.\frac{\partial X}{\partial \alpha}\right|_{t_1} \tag{A4}$$

Comparison of equation (A1) and equation (A2) shows that equation (A1) is related to the homogeneous form of equation (A2). This suggests that a variation of parameters method might be used to obtain a solution to equation (A2).

Assume equation (A2) has a solution of the form

$$\left.\frac{\partial X}{\partial \alpha}\right|_{t_o} = \frac{\partial X}{\partial X_o} Q(t) \quad ; \quad Q(t_o) = \hat{0}$$

where Q is the matrix of parameters to be determined. Substitution of this equation in equation (A2) and using equation (A1) gives

$$\frac{\partial X}{\partial X_o}\dot{Q} = \left.\frac{\partial \dot{X}}{\partial \alpha}\right|_{t}$$

$$Q(t) = \int_{t_o}^{t} \frac{\partial X_o}{\partial X_1}\left.\frac{\partial \dot{X}_1}{\partial \alpha}\right|_{t_1} dt_1$$

By using equation (A3), the solution to equation (A2) is expressed by the quadrature

$$\left.\frac{\partial X}{\partial \alpha}\right|_{t_o} = \int_{t_o}^{t} \frac{\partial X}{\partial X_1}\left.\frac{\partial \dot{X}_1}{\partial \alpha}\right|_{t_1} dt_1 \tag{A5}$$

Note the following properties of the integrand in equation (A5):

a. If the perturbing forces (noncentral-body forces) are small, a good approximation for $\partial X/\partial X_1$ may be computed from a mean conic reference defined by the known endpoint conditions, X, t and X_1, t_1 (ref. 4 and 5).

b. The term $\left.\frac{\partial X_1}{\partial \alpha}\right|_{t_1}$ has the form

$$\left.\frac{\partial \dot{X}_1}{\partial \alpha}\right|_{t_1} = \begin{bmatrix} \hat{0} \\ \hline \left.\frac{\partial A_1}{\partial \alpha}\right|_{t_1} \end{bmatrix}$$

$\hat{0}$ = 3xM zero matrix

A_1 = Vehicle acceleration at time t_1

This term is known since the models for the perturbing forces are known.

A stepwise numerical quadrature for equation (A5) may be devised as follows. Let t_n and t_{n-1} be the endpoint times in equation (A5). Trapezoid integration yields

$$\left.\frac{\partial X_n}{\partial \alpha}\right|_{t_{n-1}} = \int_{t_{n-1}}^{t_n} \frac{\partial X(t_n)}{\partial X(t')} \left.\frac{\partial \dot{X}(t')}{\partial \alpha}\right|_{t'} dt'$$

$$= \frac{t_n - t_{n-1}}{2} \left[\frac{\partial X_n}{\partial X_{n-1}} \left.\frac{\partial \dot{X}_{n-1}}{\partial \alpha}\right|_{t_{n-1}} + \left.\frac{\partial \dot{X}_n}{\partial \alpha}\right|_{t_n} \right] \tag{A6}$$

Equations (A3) and (A4) are used to express

$$\frac{\partial X_n}{\partial X_o} = \frac{\partial X_n}{\partial X_{n-1}} \frac{\partial X_{n-1}}{\partial X_o} \tag{A7}$$

$$\left.\frac{\partial X_n}{\partial \alpha}\right|_{t_o} = \frac{\partial X_n}{\partial X_{n-1}} \left.\frac{\partial X_{n-1}}{\partial \alpha}\right|_{t_o} + \left.\frac{\partial X_n}{\partial \alpha}\right|_{t_{n-1}} \tag{A8}$$

The initial values at, t_o for these computations are

$$\frac{\partial X_o}{\partial X_o} = I \text{ , 6x6 unit matrix}$$

$$\left.\frac{\partial X_o}{\partial \alpha}\right|_{t_o} = \hat{0} \text{ , 6xM zero matrix}$$

Equations (A6), (A7), and (A8) are recognized as the key equations used in section 3.2 of this document.

APPENDIX B

FLOW CHARTS FOR STMM MECHANIZATION

APPENDIX B

FLOW CHARTS AND INTERFACE TABLES

BI. FLOW CHARTS FOR STMM MECHANIZATION

The flow charts contained in this appendix present a particular mechanization of the functional requirements given in the text. They are included only as an aid to assist in the understanding of the functional requirements. This does not imply that the mechanization shown is the most efficient for the real-time program.

B2. INTERFACE TABLES

Interface tables for the STMM are also contained in this appendix.

The following are additional notes.

B3. SUPPLEMENTAL NOTES

a. Inputs to STMM come from DCM or CP and the output from STMM is the state transition matrix π. Maximum dimension possible is 6 by 15.

b. The BϕDATT routine is defined in the level B and C requirements for the free-flight predictor and requires an input altitude table that is formed by a preprocessor for time points throughout the integration interval. The output is the matrix transformation from body coordinates to M50 coordinates at each integration time point.

c. Inputs to TMATRIX are from DCM or CP via STMM. They are as follows:

   (1) TϕBS : Time of observation, t
   (2) TLAST: Initial M50 epoch
   (3) VECTL: M50 state (initial) at t_L
   (4) VECT : M50 state at output time t

   Output from TMATRX is the state transition matrix obtained by using mean conic partial derivative function.

TABLE BI.- DEFINITION OF VARIABLES USED IN THE FLOW CHART

Notation used in flow chart	Notation used in level C requirements	Notation used in flow chart	Notation used in level C requirement
AI3	I_3	E1,E2,...,E8,E9	Temporary variable
ALPHA	α	F	f
C(6)	$C_5(m)$	FDϕT	$\dot{f}$
C(5)	$C_4(m)$	G	g
		GDϕT	$\dot{g}$
C(4)	$C_3(m)$	PSI	ψ
C(3)	$C_2(m)$	RDϕT	$\dot{R}$
C(2)	$C_1(m)$	RDDϕT	$\ddot{R}$
C(1)	$C_0(m)$	RDDϕTO	$\ddot{R}_o$
CMU	μ	S1	S_1
DO	D_o	S2	S_2
DELROO	RR_o^T	S3	S_3
DELRO1	$R\dot{R}_o^T$	SR	r
DELRO2	$\dot{R}R_o^T$	SRO	r_o
DELRO3	$\dot{R}\dot{R}_o^T$	ST	t
DELRO4	$\ddot{R}\ddot{R}_o^T$	STO	t_o
DELRO5	$\ddot{R}\dot{R}_o^T$	T	$T(t,t_o)$
DELRO6	$\dot{R}\ddot{R}_o^T$	VOSQ	V_o^2
DRDRO	$\partial R/\partial R_o$	VEC	X
DRDRDO	$\partial R/\partial \dot{R}_o$	VECMNI	X_o
DRDDRO	$\partial \dot{R}/\partial R_o$	VSQ	V^2
DRDDRD	$\partial \dot{R}/\partial \dot{R}$	X2	χ^2

TABLE BII.- DEFINITIONS OF CONSTANTS AND FLAGS

Notation	Definition
BTAB	Array of attitude information defined chronologically (ref. 8)
CDA(I) CDF(I) CDN(I) CDS(I) N(I)	Constants for ATTITUDE-DEPENDENT DRAG model for each configuration I = 1 Shuttle payload bay doors closed I = 2 Shuttle payload bay doors open
CMU	Gravitational parameter of Earth alone (E.r.^{3}/hr^2)
DRAG	(KC_DA) Flag to specify atmospheric drag calculation mode (ref. 8) Note: The values are derived from the integrator force options. The values of DRAG are as follow: >0 Attitude-independent drag (contains the numerical value of the product of the drag correction factor, the coefficient of drag, and the cross-section area) =0 No drag <0 Attitude-dependent drag; tabular input coefficients required
GVENT	Conversion factor $\frac{\text{lb} \quad \text{E.r./hr}^2}{\text{slg} \quad \text{ft/sec}^2}$
NBIAS	Total number of biases in the solution vector
NVNTSL	Number of vents in the solution vector, can take values 0, 1, 2 or 3; maximum of three vents is possible (no drag) if NVNTSL = 0, then no vents are solved
VSϕL	Total number of parameters in the solution vector, maximum can be 15 including the biases, vents, drag and state
RNP	RNP matrix is obtained from the system
DTAB	Vehicle weight and configuration table specified chronologically
IBATT	Set to 0 if body attitude matrix is not needed Set to 1 if body attitude matrix is to be computed
ISLUDR	Set to 0 if DRAG-MULTIPLIER is not in solution vector Set to 1 if DRAG-MULTIPLIER is in solution vector

TABLE BII.- DEFINITIONS OF CONSTANTS AND FLAGS - Concluded

Notation	Definition
ω_E	Angular rate of rotation of the Earth (E.r./hr)

```
 _____       _____
!     !  to !     !
! DCM !---->! STMM !  INTERFACE TABLE*
!_____!     !_____!
```

DCM parameter[a]	STMM parameter[b]	Unit	Description
Link ID	Link ID	Flag	Identifies DC link
$\vec{R}_L, \vec{V}_L, t_L$	X_L, t_L	Internal	M50 state and epoch at initialization time
$\vec{R}, \vec{V}, t$	X, t	Internal	M50 state and epoch at output time
$\Phi(t_L, t_o)$	$\Phi(t_L, t_o)$	Internal	State transition matrix that maps from anchor time (t_o) to initialization time (t_L)
SVFLGS	Solution vector content	Flag	Identifies "solve-for" dynamic parameters and biases in the solution vector.
VNT(J)	VNT_J	Flag	Specifies whether Jth vent is ON or OFF for current computation
	Integrator force options		Identifies drag model

```
 ______       _____
!      !  to !     !
! STMM !---->! DCM !  INTERFACE TABLE*
!______!     !_____!
```

STMM parameter[b]	DCM parameter[a]	Unit	Description
$\phi(t, t_o)$	$\phi(t, t_o)$	Internal	State transition matrix, 6x(6+M) where M = number dynamic parameters, which maps from anchor time (t_o) to current time (t)

[a] See table I of volume VII.
[b] See section 3.2 of this document.

*STMM/CP Interface is identical

CMP to STMM INTERFACE TABLE

CMP parameter[a]	STMM parameter[b]	Unit	Description
X_E, t_E	X_o, t_o	Internal	M50 state and epoch at input time
A, t_A	X, t	Internal	M50 state and epoch at output time

STMM to CMP INTERFACE TABLE

STMM parameter	CMP parameter	Unit	Description
$T(t, t_o)$	$T(t_A, t_E)$	Internal	State transition matrix (6x6) that maps from input time ($t_o = t_E$) to output time ($t = t_A$)

[a]See section 3.2.3 of volume IX.
[b]See section 3.1 of this document.

```
 _______         ___________________
! STMM  ! to    !      Vehicle       !   INTERFACE TABLE
!       !---->! Characteristics    !
!_______!       !___________________!
```

STMM parameter[a]	Vehicle Characteristics parameter	Unit	Description
t		Internal	Time for required parameters
Link ID		Flag	Identifies vehicle

```
 ___________________        _______
!      Vehicle      ! to   ! STMM  !   INTERFACE TABLE
! Characteristics   !---->!       !
!___________________!      !_______!
```

Vehicle Characteristics parameter	STMM parameter[a]	Unit	Description
	W	lb	Vehicle mass
	B(t)	Internal	Transformation matrix (3x3) from vehicle body axes to M50
	$(K_D C_D S_A)$	ft^2	Product of K_D = drag multiplier, C_D = drag constant, and S_A = vehicle reference area

[a]See section 3.2 of this document.

```
!‾‾‾‾‾‾‾‾! to !‾‾‾‾‾‾‾‾‾‾‾‾!
! STMM !---->! Solar      !
!________!     ! Ephemeris !
               !____________!
```

INTERFACE TABLE

STMM parameter[a]	Solar ephemeris parameter	Unit	Description
t		Internal	Time at which solar position is desired

```
!‾‾‾‾‾‾‾‾‾‾‾‾! to !‾‾‾‾‾‾‾‾!
!   Solar    !---->! STMM !
! Ephemeris !      !________!
!____________!
```

INTERFACE TABLE

Solar ephemeris parameter	STMM parameter[a]	Unit	Description
	$\vec{r}$(SUN)	Internal	Solar position (M50) at required time, t

[a]See section 3.2 of this document.

```
 ______        ___________
!      !  to  !           !
! STMM !----->! Density   !   INTERFACE TABLE
!______!      ! Module    !
              !___________!
```

STMM parameter[a]	Density module parameter	Unit	Description
t		Internal	Time at which density is required
$\vec{r}$		Internal	Vehicle position (M50) at time t
$\vec{r}$(SUN)		Internal	Solar position (M50) at time t

```
 ___________        ______
!           !  to  !      !
! Density   !----->! STMM !   INTERFACE TABLE
! module    !      !______!
!___________!
```

Density module parameter	STMM parameter[a]	Unit	Description
	$\rho(t)$	slg/ft^3	Atmospheric density of vehicle at time t

[a]See section 3.2 of this document.

STMM --to--> RTRANS INTERFACE TABLE

STMM parameter	RTRANS parameter	Unit	Description
t		Internal	Time for which (RNP) is desired

RTRANS --to--> STMM INTERFACE TABLE

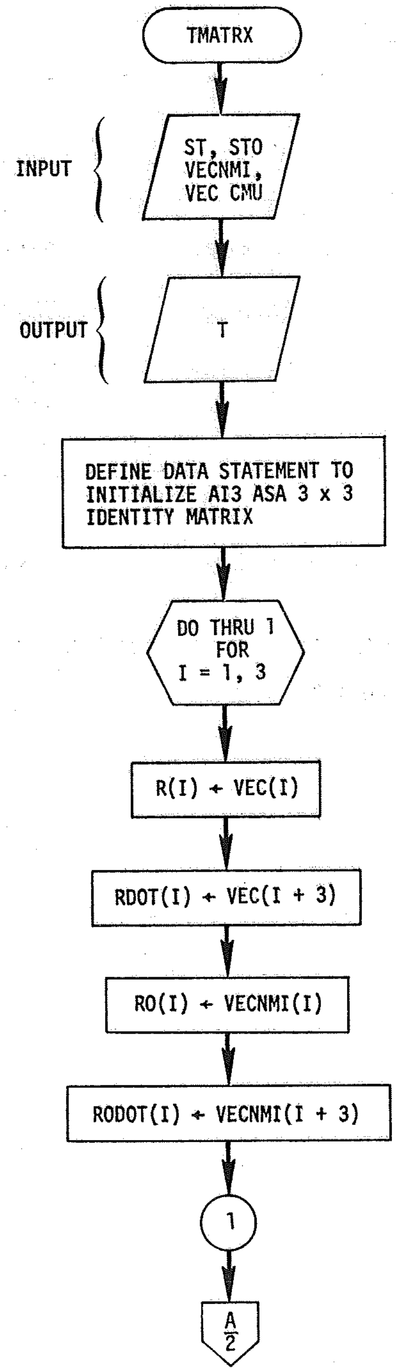
RTRANS parameter	STMM parameter	Unit	Description
	(RNP)	Internal	RNP-matrix for desired time

[a]See section 3.2 of this document.

System parameter $\xrightarrow{\text{to}}$ STMM INTERFACE TABLE

System parameter	STMM parameter[a]	Unit	Description
	μ	Integer	Earth gravitational parameter
	g_O	lb/slg	Standard gravity
	w_E	Integer	Earth spin rate (mean sidereal)
	(GVENT)	$\frac{\text{lb}\cdot\text{E.r./hr}^2}{\text{slg}\cdot\text{ft/sec}^2}$	Conversion factor
	(FEET)	ft/E.r.	Conversion factor

[a]See section 3.2 of this document.



Page 1 of 6

Figure 1.- Flowchart to compute $T(t, t_0)$ conic state partial derivative function.

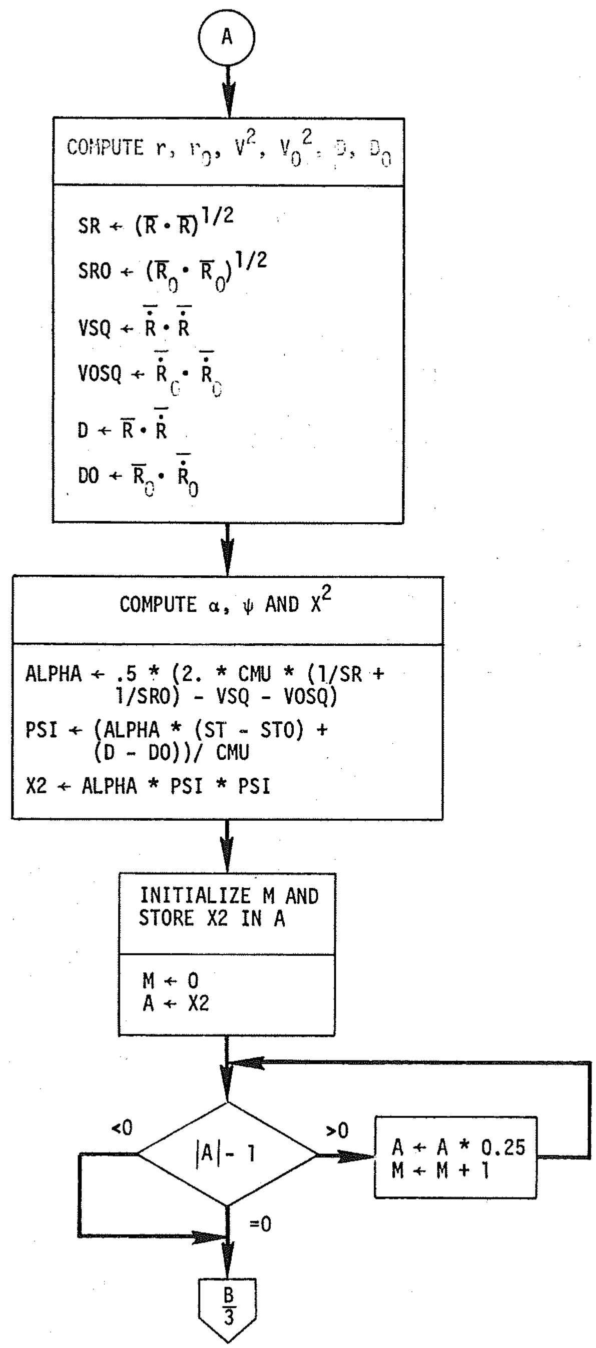
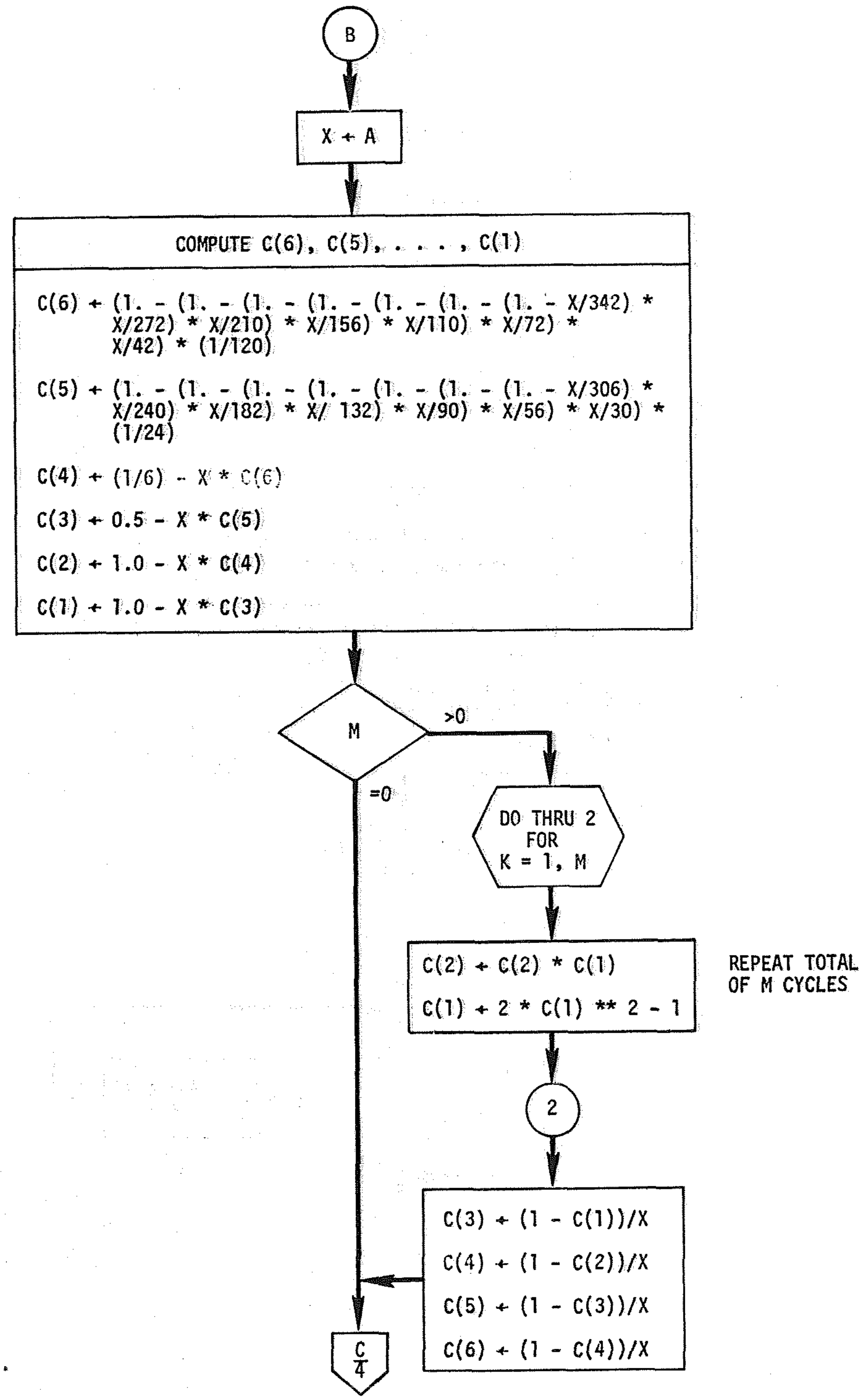


Figure 1.- Continued.



Page 3 of 6

Figure 1.- Continued.

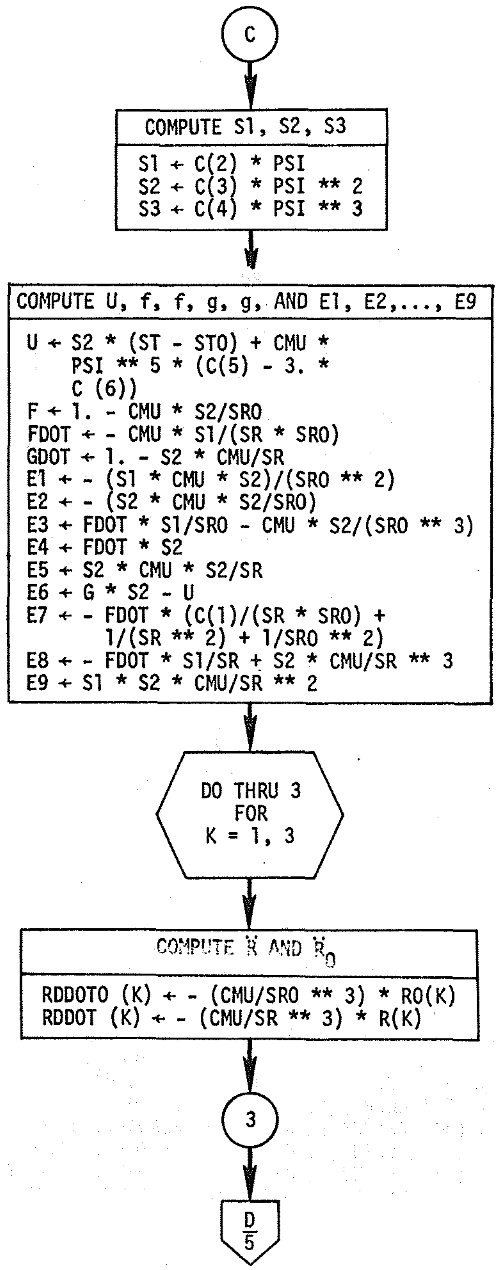


Figure 1.- Continued

(D)

COMPUTE RR_0^T, $R\dot{R}_0^T$, $\dot{R}R_0^T$, $\dot{R}\dot{R}_0^T$, ETC

DELR00 ← $\overline{RR_0^T}$

DELR01 ← $\overline{R\dot{R}_0^T}$

DELR02 ← $\overline{\dot{R}R_0^T}$

DELR03 ← $\overline{\dot{R}\dot{R}_0^T}$

DELR04 ← $\overline{R\dddot{R}_0^T}$

DELR05 ← $\overline{\ddot{R}R_0^T}$

DELR06 ← $\overline{\dot{R}\ddot{R}_0^T}$

DO THRU 4
FOR
I = 1, 3

DO THRU 4
FOR
J = 1, 3

COMPUTE $\partial R/\partial R_0$, $\partial R/\partial \dot{R}_0$, $\partial \dot{R}/\partial R_0$ AND $\partial \dot{R}/\partial \dot{R}_0$

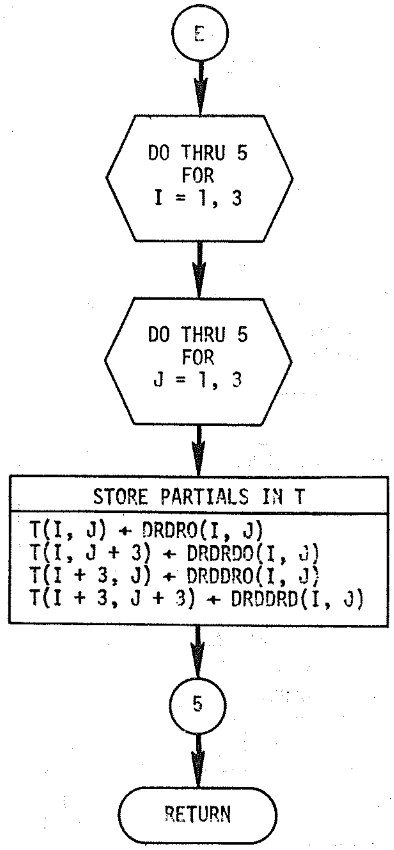
```
DRDRO(I, J) ← - E3 * DELR00(I, J) - E4 * DELR01(I, J) + E1 * DELR02(I, J)
            + E2 * DELR03(I, J) + U * DELR06(I, J) + F * AI3(I, J)
DRDRDO(I, J) ← - E4 * DELR00(I, J) + E5 * DELR01(I, J) + E2 * DELR02(I, J)
             + E6 * DELR03(I, J) + G * AI3(I, J)
DRDDRO(I, J) ← E7 * DELR00(I, J) + E8 * DELR01(I, J) + E3 * DELR02(I, J)
             + E4 * DELR03(I, J) + U * DELR04(I, J) + FDOT * AI3(I, J)
DRDDRD(I, J) ← E8 * DELR00(I, J) + E9 * DELR01(I, J) + E4 * DELR02(I, J)
             - E5 * DELR03(I, J) - U * DELR05(I, J) + GDOT * AI3(I, J)
```

(4)

E/6

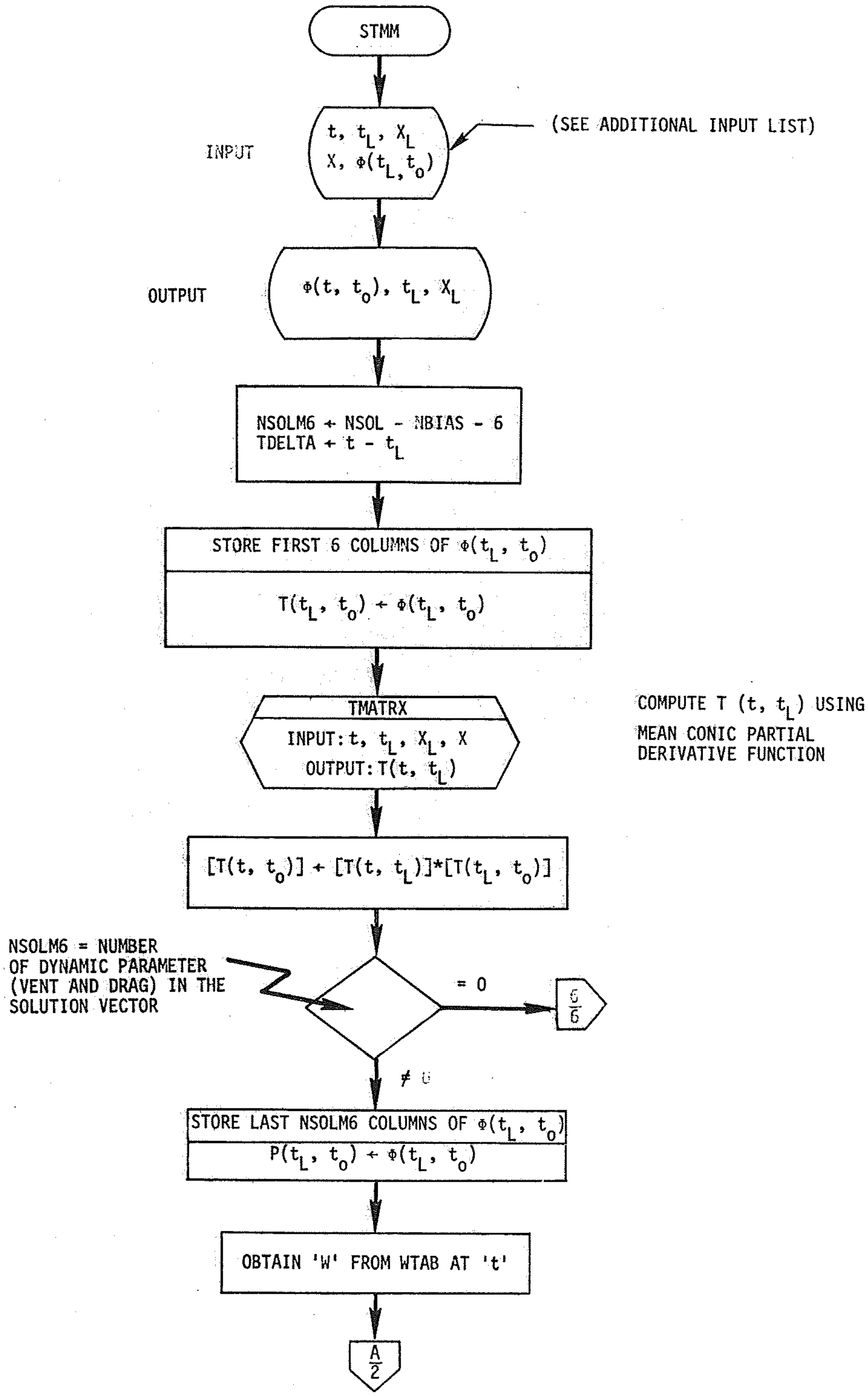
Page 5 of 6

Figure 1.- Continued



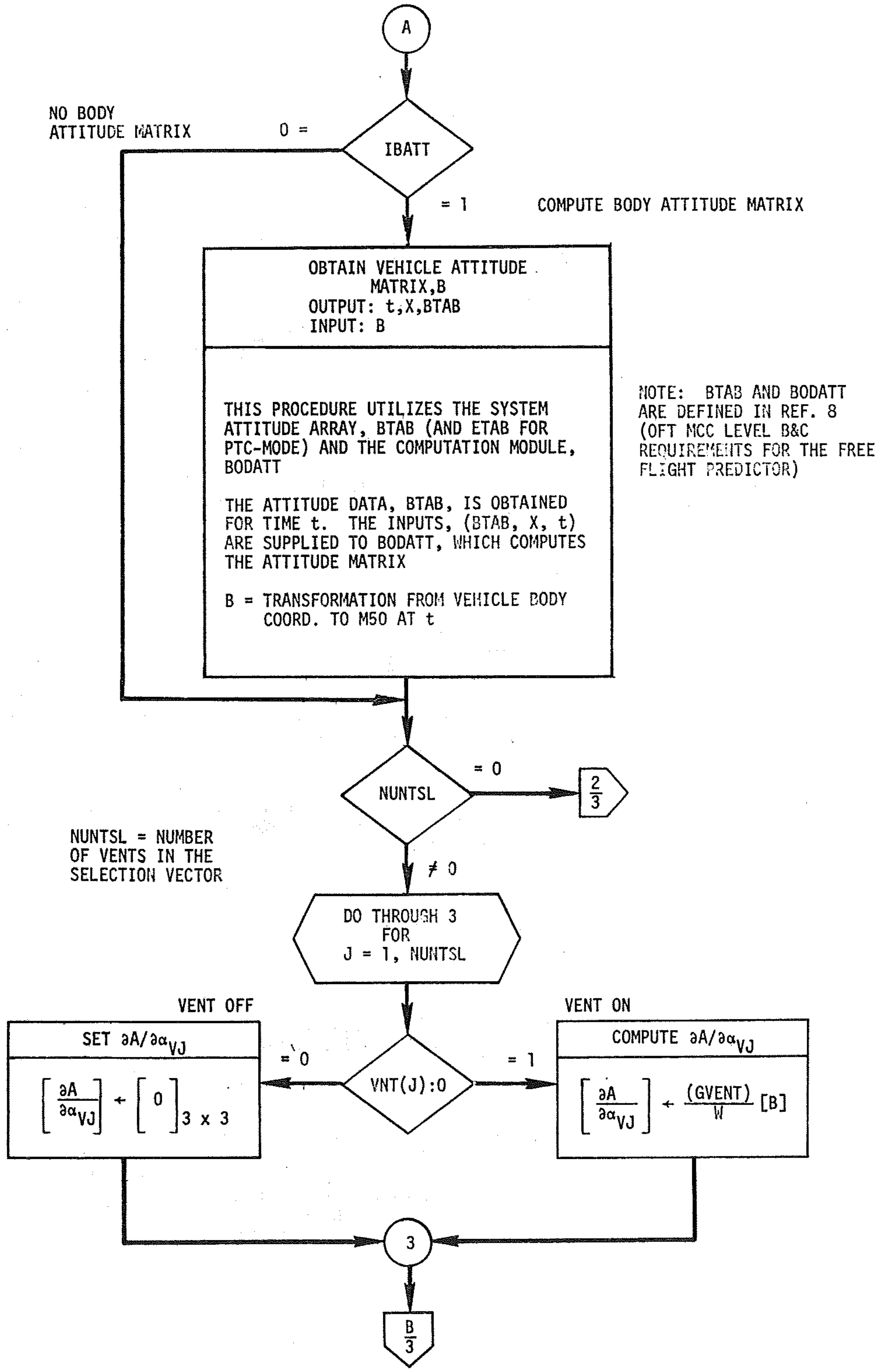
Page 6 of 6

Figure 1.- Concluded



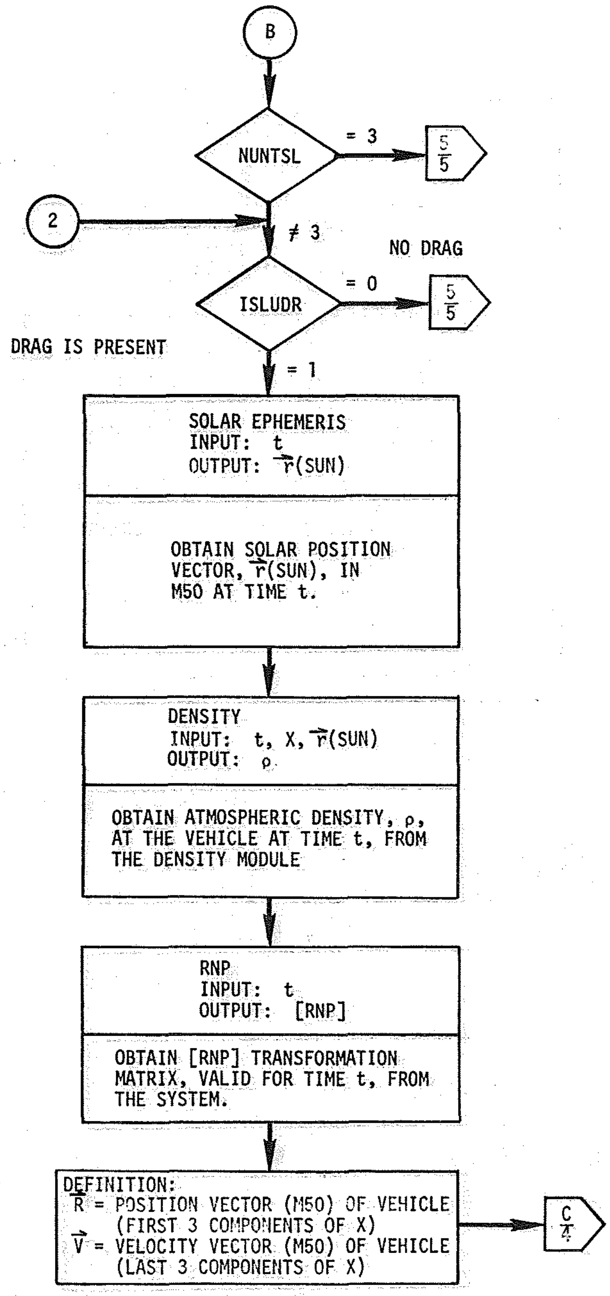
Page 1 of 6

Figure 2.- Flowchart to compute state transition matrix via state transition integrator.



Page 2 of 6

Figure 2.- Continued.



Page 3 of 6

Figure 2.- Continued.

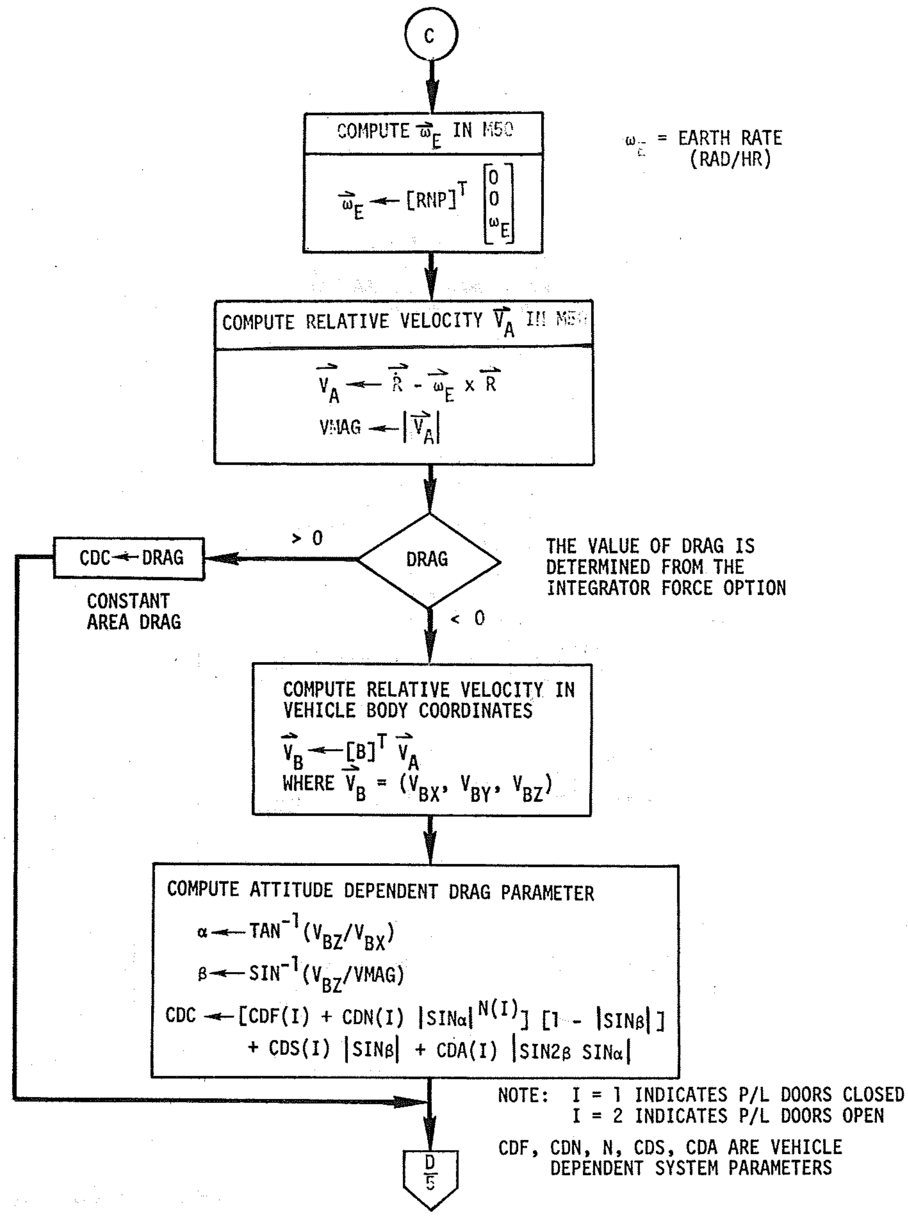


Figure [illegible].- Continued.

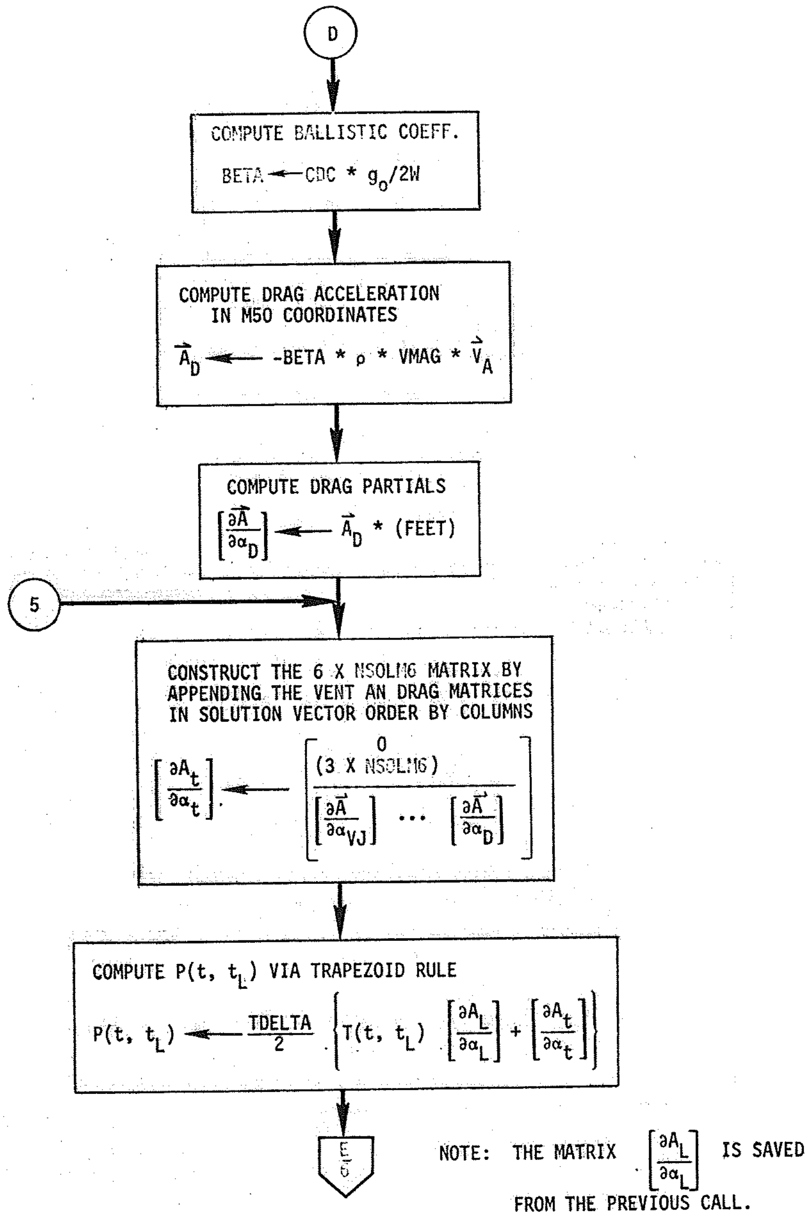
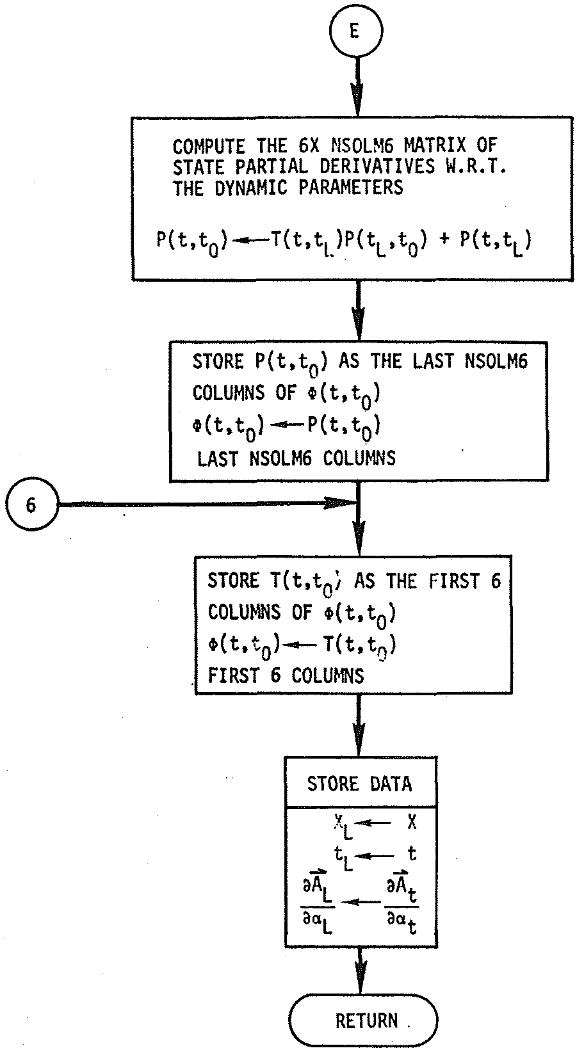


Figure ?.- Continued.



Page 6 of 6

Figure 2.- Concluded.

☆ U.S. GOVERNMENT PRINTING OFFICE: 1979—671-014/22 19

DISTRIBUTION FOR JSC IN 78-FM-30, VOL. X

JM6/Technical Library (2)
JM61/Center Data Management (3)
CF3/J. Greene
FM8/E. Schiesser
    J. Williamson (5)
    J. Currie
    W. York
    J. Weaver
    W. Wollenhaupt
    R. Osburn
FM14/Report Control (20)
    A. Wiseman
    B. Woodland
FM17/L. Hartley
FS5/M. Dixon
    J. Mendiola
FS15/R. Brown (2)
IBM/H. Norman
    R. Rich
    W. Goodyear
    F. Riddle
    C. Waund
    A. Stevenson (8)
MDTSCO/T. Rich
    R. Theis
TRW/O. Bergman (15)